

CAD TO GEANT4/ROOT GEOMETRY BUILDER. A TOOL FOR LHC AND FAIR EXPERIMENTS SYSTEM ENGINEERING

**Belogurov Sergey⁽¹⁾, Berchun Yuri⁽²⁾, Chernogorov Andrey⁽¹⁾, Malzacher Peter⁽³⁾,
Ovcharenko Egor^(1, 2), Semennikov Alexander⁽¹⁾**

*⁽¹⁾Institute for Theoretical and Experimental Physics, B. Cheremushkinskaja 25, 117218,
Moscow, Russia. belogurov@itep.ru*

*⁽²⁾Bauman Moscow State Technical University, 2-nd Baumanskaja str. 5, 105005,
Moscow, Russia, cad@bmstu.ru*

*⁽³⁾GSI Helmholtzzentrum für Schwerionenforschung GmbH, Planckstr. 1, 64291,
Darmstadt, Germany, P.Malzacher@gsi.de*

INTRODUCTION

System Engineering (SE) of complex densely packed installations requires careful multiparametric optimization. One should take into account not only usual mechanical, thermal and electromagnetic mutual influence of the subsystems, but also specific for a particular case effects.

Big experiments in High Energy Physics give an example of such complex systems. Except for common layout and mounting issues, each subdetector or service subsystem is perceivable by others in terms of cross talks, primary and secondary radiation shielding, and generation of background. Optimization of a subdetector is also a complicated task. For example, increasing the number and decreasing the size of independent sensors (raising granularity) one improves resolution of the detector and simplifies intellectual data processing (like finding of particles trajectories), on the other hand, the raw data flow, amount of dead material in the readout chains, and price dramatically grow up. It is a non-trivial SE problem to decompose tasks and optimize the design of a modern experiment at big accelerators.

For optimizing the experimental setup one has to simulate physical processes accompanying ionizing and nuclear radiation. Primarily such Monte Carlo (MC) simulation tools were developed for research and applications involving radioactivity, accelerated beams, and nuclear reactors. However, similar simulations are also necessary for the space systems due to extraterrestrial radiation environment.

It is practically impossible to finalize the design without multiple data exchange between Computer Aided Design (CAD) systems and radiation MC software. This iterative loop is especially important and labor-intensive for modern experiments due not only to their complexity, but also to high sensitivity to wrong solutions in the severe radiation environment with huge number of secondary particles and high interaction rates.

One can routinely transfer mathematical models (i.e. geometry and material description) of the objects under investigation between CAD systems and mechanical, thermal, and electromagnetic simulation software. In contrary, geometry representation in radiation MC packages widely used in radiation studies is rather different. Here we focus on the most diffused in the field of Particle and Relativistic Nuclear Physics open source software GEANT4 (G4) [1]. The same geometry representation is independently implemented in the ROOT package [2].

For exchanging G4/ROOT-compatible geometry, Geometry Description Markup Language (GDML) [3] is used.

Traditionally, geometry for MC simulations is quite simple at the design and optimization stage. It reflects not detailed, but conceptual layout of the setup. When design is finished, the detailed engineering drawings are used for manual creation of more precise geometry description. The last operation is very laborious and can hardly be repeated several times. However, for modern experiments the design optimization process should be iterative.

In principle, it is possible to generate automatically a tessellated solid from any CAD model. Produced geometry can be imported then into G4/ROOT. Such an approach is realized e.g. in the FASTRAD package [4]. However, due to slow simulation speed for big assemblies and complicated shapes, direct automated transfer via tessellated solids has limited applicability.

We are presenting a set of tools, which allows to facilitate significantly creation of G4/ROOT-compatible geometry from the CAD system CATIA v.5. This system is widely used in big physical laboratories like CERN, GSI, LNGS, and other scientific and industrial entities. The tools are being developed for upcoming upgrade of LHC experiments and for experiments at the future Facility for Antiproton and Ion Research (FAIR), however, both systems (CATIA v.5 and GEANT4) are used in the space applications as well [5].

Thus, we are bringing an important link into the SE chart of the systems, affected by radiation.

GEOMETRY REPRESENTATION IN CAD AND GEANT4/ROOT

Very important input for the reported development is a big difference between the geometry representations in the CAD systems and G4/ ROOT. The difference is twofold: in the description of solid bodies and in the hierarchy of assemblies.

Geometry of a Solid

In CAD systems solid bodies are built using a wide class of surfaces and curves in advanced boundary representation (BRep).

In G4/ROOT the Constructive Solid Geometry (CSG) is used. Fundamental building blocks for CSG are so called primitives - 3D objects described by a minimal set of parameters necessary to define shape and size of the solid. Currently implemented set of primitives includes 21 items and ranges from simple box, tube and sphere to rather complicated twisted trapezoid and polyhedra. Some of the primitives are shown in the fig. 1.

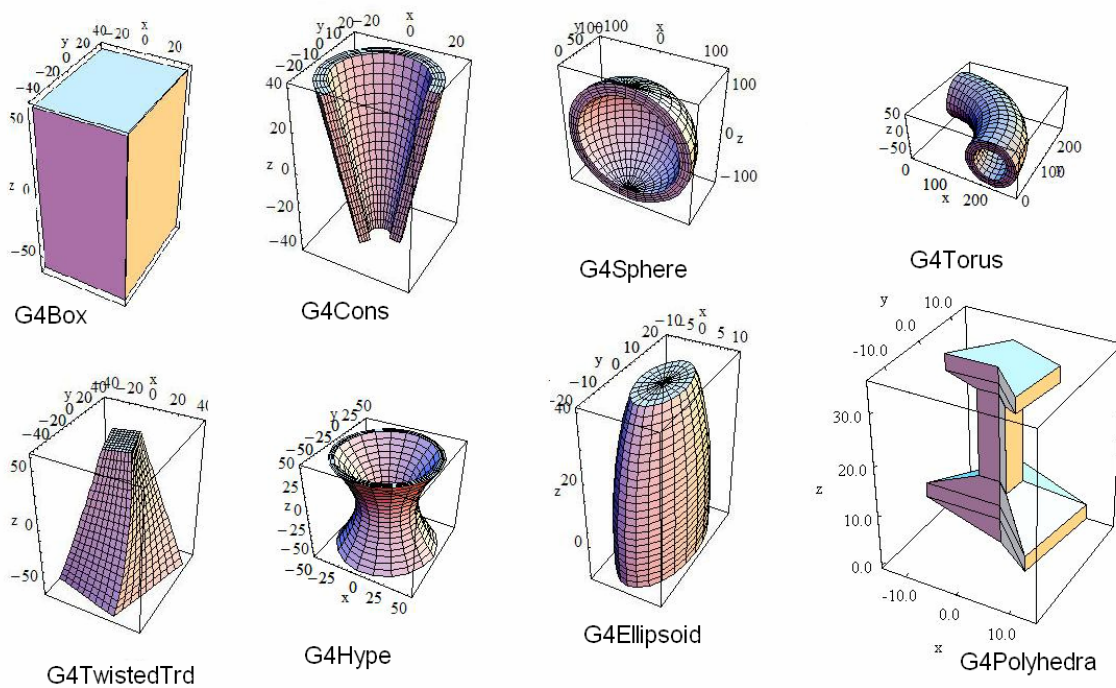


Fig. 1. Some of G4 primitives.

Simple solids can be combined using Boolean operations. Creating such a new Boolean solid requires: two solids (operands), a Boolean operation: *union*, *intersection* or *subtraction*, optionally a transformation (*scale*, *rotation*, and *translation*) for the second solid.

An operand for a Boolean operation should be either primitive or another Boolean solid: the result of a previous Boolean operation. An important purpose of Boolean solids is to enable description of solids with peculiar shapes in a simple and intuitive way, still allowing an efficient geometrical navigation inside them. The final Boolean solid should represent a single 'closed' solid, i.e. a Boolean operation between two solids, which are disjoint is not a valid Boolean composition.

Assembly Hierarchy

In CAD software there is a minimal indivisible unit – a solid body (*part*) and conceptually unlimited number of nested assemblies (*products*). Products and subproducts are only logical units – all the materials are assigned to solid bodies inside the part files or to parts themselves.

In G4/ROOT detector geometry is made of a number of volumes. The largest one is called the *World*. It contains all other volumes in the detector geometry. The other volumes are created and placed inside previous (mother) volumes, included in the *World* volume. Each volume is created by describing its shape and its physical characteristics, and then placing it inside a containing volume. To describe a volume's shape, the concept of a *solid* is used. To describe a volume's full properties, one uses a *logical volume*. It includes the geometrical properties of the solid, and adds physical characteristics: material of the volume, whether it contains any sensitive detector elements, magnetic field, etc. To define a position of the volume one should create a physical volume – a copy of the *logical volume* put inside a larger containing volume. Important is that unlike a *part* in the CAD systems, any logical volume may be a mother for placing smaller volumes inside. The *World* is a special *logical volume*, because it can not be positioned.

Beside very specific method of geometry description and product structure for G4/ROOT, there are some limitations from the side of computational resources. Preparing any geometry for MC simulations one has to keep in mind other two issues:

Required Detalization

G4/ROOT model requires normally only details perceivable by the detectors. For a detector with poor position resolution, description should be less accurate than for more precise device. Choosing appropriate level of detalization, one can reduce significantly required computational time and memory. E.g. threaded hole in a flange with a screw inside is equivalent for MC to a bulk piece of metal; for peripheral equipment, sometimes, just a simple solid filled with a correct mixture of materials is sufficient. For CAD models, instead, all the details are essential.

Simulation Optimization

There are various tricks allowing to optimize simulations and reduce geometry description file. Here is a couple of examples.

Description of a sampling calorimeter [6] will be more compact and loading of the geometry faster if scintillator plates are inserted into the bulk lead mother volume, fig. 2. In CAD assembly one always has a stack of alternating lead and plastic plates, fig. 3.

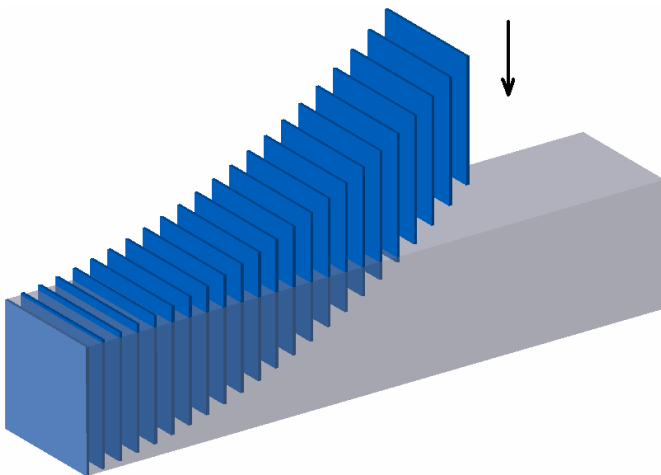


Fig. 2. G4/ROOT-like representation of a module of the sampling calorimeter

Putting several simple solids of the same material next to each other instead of using union helps to accelerate simulations. As an example, CAD and G4/ROOT representations of a section of the beampipe for CBM experiment at FAIR are shown in the fig. 4.

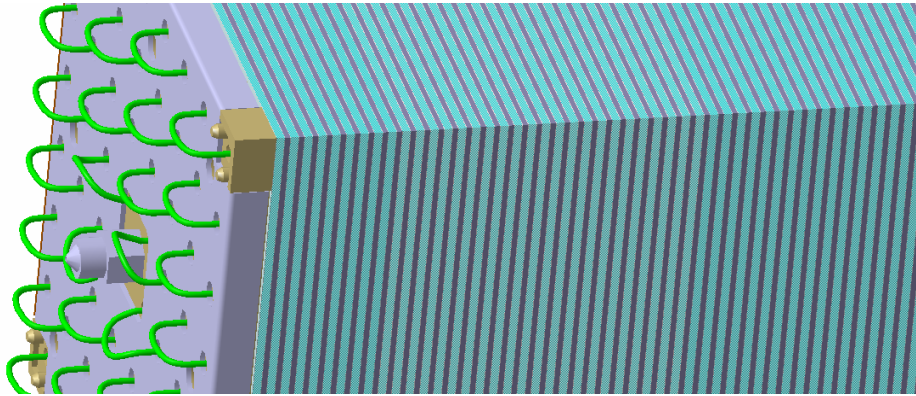


Fig. 3. CAD representation of a module of the sampling calorimeter

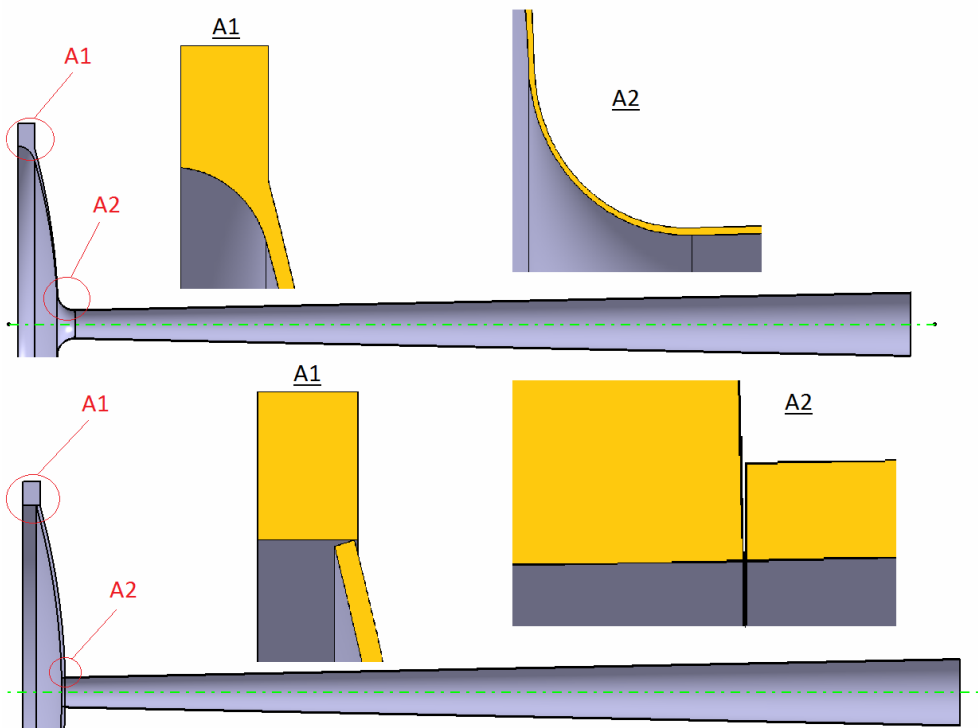


Fig. 4. CAD and G4/ROOT representations of the CBM beampipe. G4 representation consists of a tube, a sphere, and a cone.

AN APPROACH TO G4 GEOMETRY BUILDING FROM INSIDE CATIA

The goal of the CATIA-GDML geometry builder is to make manual geometry transfer from CATIA to GEANT4 more convenient and fast than before. We employ powerful measurement, analysis, and design features of various CATIA workbenches to create a MC-compatible representation of an existing engineering assembly.

Manual geometry building allows to have enough flexibility for creation of the geometry optimal for fast and efficient simulation.

The tool is targeted on scientists who understand what geometry representation and level of detalization are optimal for given simulation task. For usage of the tool they should get familiar with only limited functionality of CATIA and our plugins.

Implementation of Primitives and Logical Volumes

The following approach is realized in the geometry builder. For the most widely used G4/ROOT primitives we built parameterized User Defined Features (UDF) in CATIA. UDFs for G4Cone and G4Torus primitives are shown as an example in the fig. 5.

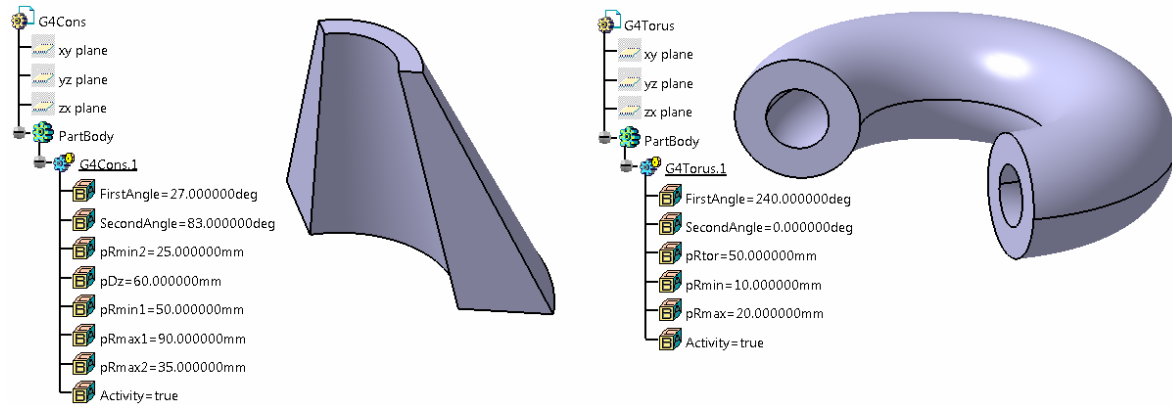


Fig. 5. CATIA realization of G4Cons and G4Torus.

We developed a data structure of the CATPart file which allows for implementation of the G4/ROOT volume hierarchy i.e. concepts of logical, physical, mother volumes, and materials. The file structure contains a *Partbody* which is a CSG solid with a material from *G4Material* catalog assigned – this stands for *logical volume*. A publication of the *Partbody* keeps neither geometry parameterizaion nor material. It serves for insertion into bigger (mother) *logical volume*. Publications of smaller volumes from other files can be inserted into CATPart file as additional bodies with positioning, it is an analogue of *physical volume*. An example of the model tree is shown in the fig. 6.

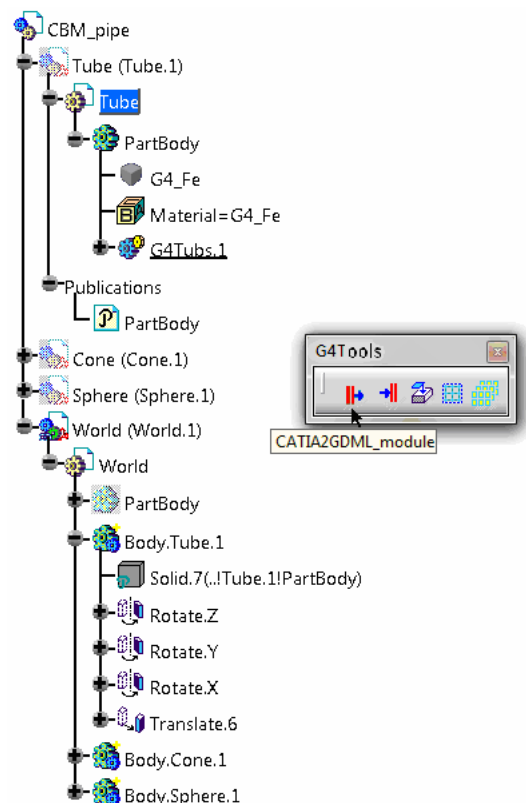


Fig. 6. G4-like structure in CATIA tree and “G4Tools” toolbar

Implementation of Physical Volumes

We developed a set of macros enabling positioning and replication of the G4/ROOT-like CAD volumes in the corresponding mother volumes, note *G4Tools* toolbar in the fig. 6. A multilevel G4/ROOT-like assembly can be built in the following way. First for each volume a *CATPart* file with parameterized solid in the *Partbody*, assigned material, and publication of the *Partbody* should be created. Then all those files should be loaded as components into a *CATProduct* file. The last step is to run inserter or replication macros for placing volumes into their mothers. Positioning template will be generated automatically. The procedure is illustrated in the fig. 7. Note bright and pale icons in the trees corresponding to shown and hidden objects.

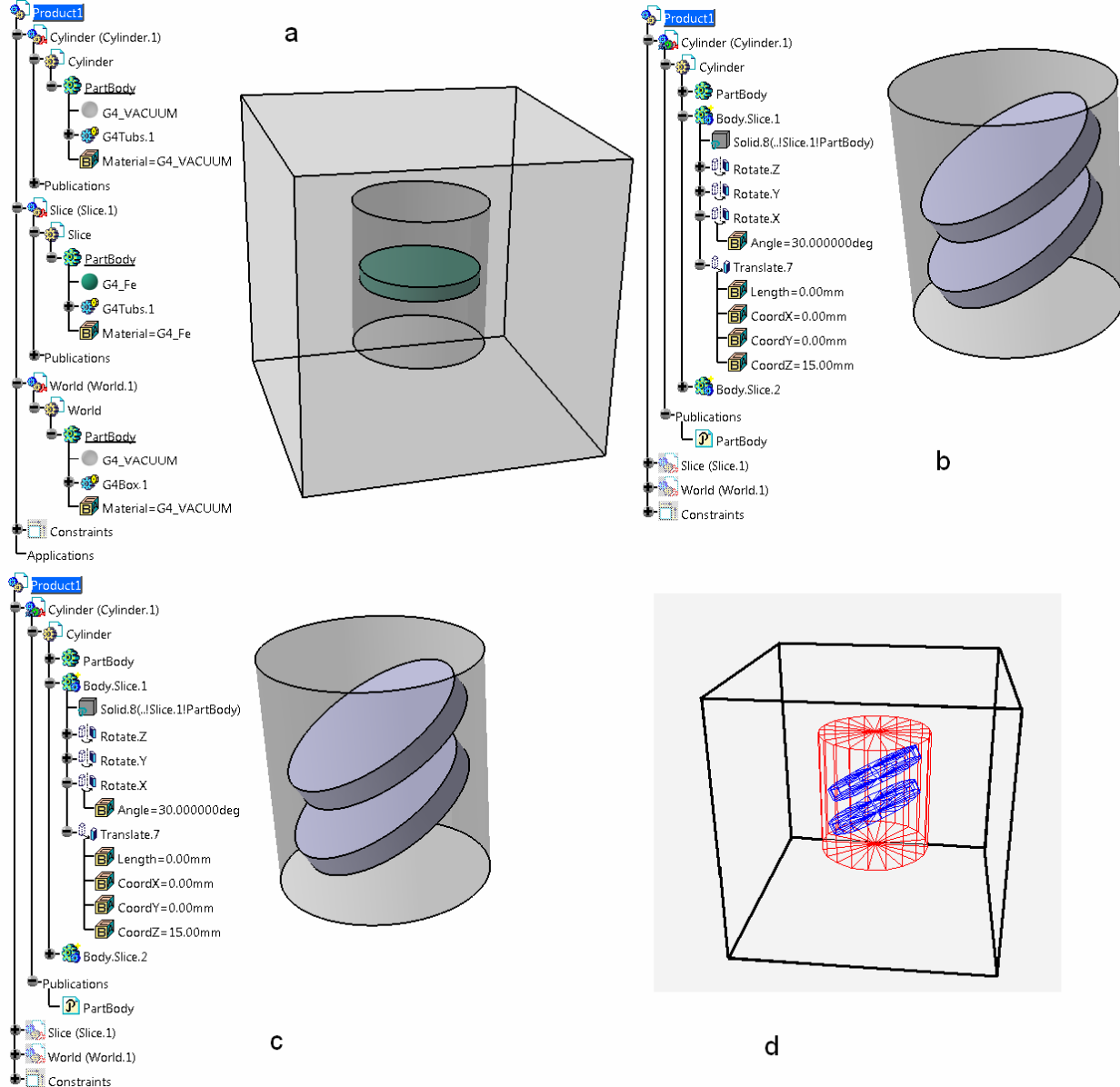


Fig. 7. Building G4/ROOT-like geometry. a) 3 Logical volumes in a product; b) inserting and positioning two slices in the cylinder; c) positioning the cylinder in the *World*; d) Exported geometry read by ROOT.

Building a G4/ROOT representation of given CAD parts and products, the user can choose the best way of a solid description and the assembly hierarchy from point of view of the simulation performance.

At the end the entire model tree is analyzed and exported into GDML. An example of geometry transferred via GDML into ROOT is shown in the fig. 7d.

In addition, we found that the opposite direction which makes the design loop closed is also requested by the community. Following that demand we implemented also inverse operation i.e. import of GDML files into CATIA. This direction is useful for shifting from physical justification of a conceptual model to engineering design.

AN EXAMPLE

A Cryostat for the Large Aperture Superconducting Dipole for R3B Experiment at FAIR.

The general CAD view of the cryostat is shown in the fig. 8a. The whole model, consisting of 4 shells of complicated shape and internal ribs, has been transferred into MC using our tool. Here we consider only one of stainless steel shells. CAD model and two G4/ROOT representations of this shell are shown in the figs. 8b-8d. Note, that in the G4/ROOT-like models there are no unions in the tree. Especially interesting is the region marked with a red oval in the fig. 8b. Here the surface between the cylinder and the torus can't be described by any of G4/ROOT primitives. In the fig. 8c this space is filled with a section of the *G4Ellipsoid*. One can see the gaps between neighboring parts. Another solution is shown in the fig. 8d. Here the radii of torus and cylinder and the length of cylinder are changed in such a way, that the problematic region can be filled without gaps with a piece of a sphere. From point of view of the radiation MC the last solution is preferable.

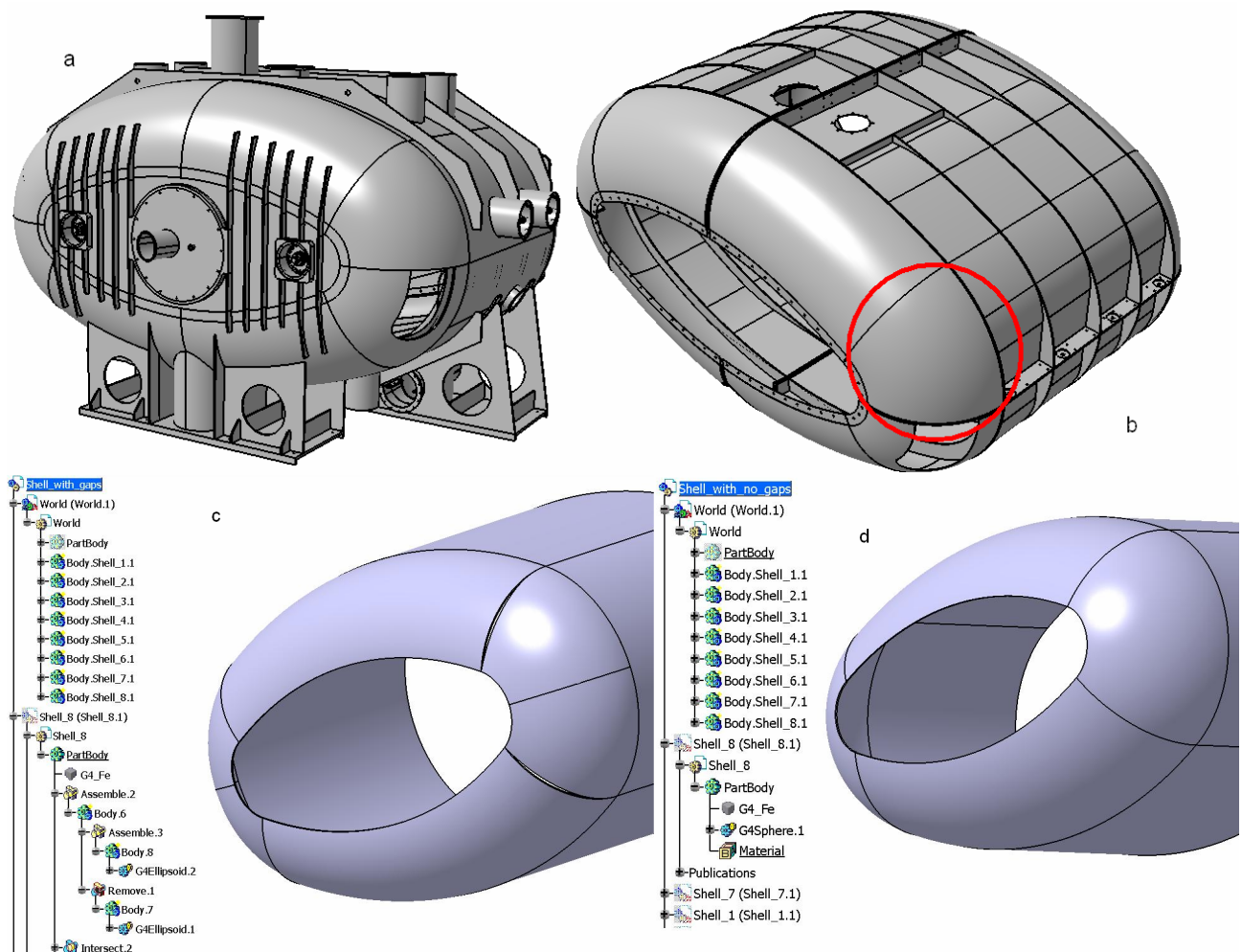


Fig. 8. a) General view of a cryostat for the R3B magnet; b) one of the Shells; c) G4 representation of the shell with gaps; d) G4 representation of the shell without gaps;

PLANS FOR FURTHER DEVELOPMENT

Further development of the Geometry Builder is foreseen in the following directions:

- Enhancement of the set of implemented primitives
- Improvement of the G4Materials catalog in CATIA
- Implementation of checkers for CSG tree structure and volume overlaps
- Adaptation of the CATIA Digital Mockup (DMU) optimizer for automatic fit of parameterized CSG models to existing parts
- Case study and best practice elaboration

CONCLUSIONS

A set of tools for building simulation-optimized GEANT-like geometry from inside CATIA CAD system is developed. The geometry building process is facilitated significantly with respect to traditional approach. The geometry can be exchanged with Physical Monte Carlo via GDML files in both directions. The tools allow iterative optimization of complex systems in harsh radiation environment. The tools are developed for System Engineering of the upgrade of LHC experiments and for the experiments at future FAIR complex. The tools can be useful also for space and medical applications.

ACKNOWLEDGEMENTS

The authors are grateful to Drs. F. Carminati, R. Brun, A. Gheata (CERN), and D. Bertini (GSI) for stimulating interest and discussions and to A. Markin (BMSTU) for his contribution into the project. The work was supported by INTAS grant 06-1000012-8778 and by FAIR-Russia Research Center.

REFERENCES

- [1] J. Allison, K. Amako, J. Apostolakis, H. Araujo et al.; "Geant4 developments and applications", *IEEE Trans. on Nucl. Sci.*, Vol. 53, Issue: 1, pp: 270 – 278, Feb. 2006
- [2] A. Naumann; E. Offermann, V. Onuchin, S. Panacek, F. Rademakers, P. Russo, and M.Tadel "ROOT — A C++ framework for petabyte data storage, statistical analysis and visualization." *Computer Physics Communications*; Vol. 180, Issue 12, pp 2499-2512, December 2009.
- [3] R. Chytrcek, J. McCormick, W. Pokorski, G. Santin "Geometry Description Markup Language for Physics Simulation and Analysis Applications" *IEEE Trans. Nucl. Sci.*, Vol. 53, Issue: 5, Part 2, pp 2892-2896, August 2006.
- [4] <http://www.fastrad.net/>
- [5] <http://geant4.esa.int/>
- [6] F. Machefert, A. Martens, "Overview of the LHCb calorimeters"; *Nucl. Instrum. Meth. Vol.A617*, pp:40-44, 2010
- [7] <http://www-win.gsi.de/r3b/Documents.htm>
- [8] http://www.iso.org/iso/iso_cafe_step.htm